ABSTRACT

Leprosy, a chronic bacterial infection, can lead to nerve damage and disability if not detected early. It is caused by infection with the bacterium Mycobacterium leprae. It mainly affects the skin, eyes, nose and peripheral nerves. This project proposes an image analysis-based Leprosy Analyzer for automated detection of the disease. The analyzer leverages image processing techniques and potentially deep learning algorithms to identify characteristic signs of leprosy in skin images. Early and accurate diagnosis plays a crucial role in successful treatment and preventing nerve damage. This Leprosy Analyzer aims to contribute to a more accessible and potentially faster diagnostic tool for healthcare professionals.

Keywords: Leprosy Screening, Contour Detection, Laplacian Filter, Canny Edge Detection, Dilation, Image Analysis

Contents

List of Figures

Chapter 1 Introduction

Leprosy, also known as Hansen's disease, is a chronic infectious disease caused by Mycobacterium leprae. It affects the skin, peripheral nerves, mucosal surfaces of the upper respiratory tract, and eyes. Leprosy can occur at any age, from early childhood to old age. It is curable, and early treatment can prevent disability. The disease is transmitted through droplets from the nose and mouth during close contact with untreated cases. The global leprosy situation has improved significantly in the last forty years since the introduction of multi-drug therapy (MDT) in 1982, with the number of cases decreasing from over 5 million in the mid-1980s to less than 200,000 by the end of 2016. But still, A total of 75,394 new cases were detected during the year 2021-22, which gives an Annual New Case Detection Rate (ANCDR) of 5.09 per 100,000 population. ANCDR was 14.27 per lakh population in 2005-06 which has reduced to 10.93 per lakh population in 2009- 10 followed by 9.71 in 2015-16 and 5.52 per lakh population in 2021-22. Due to the COVID-19 pandemic, a sudden decline in case detection has been noted. ANCDR declined from 8.13 in 2019-20 to 4.56 per lakh in 2020-21. Thereafter an upward trend is noted with a 5.52 per lakh population in 2021-22.

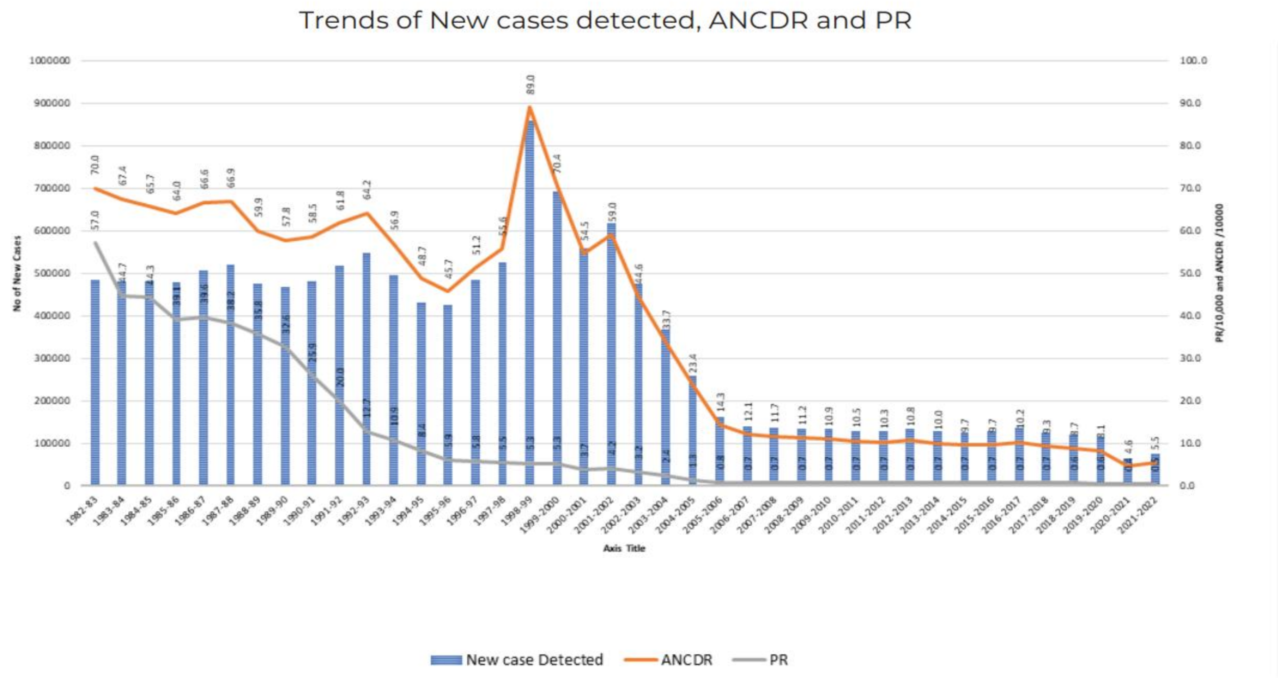

Figure 1

Leprosy remains a public health challenge in many parts of the world, particularly in developing countries with limited access to health care services. Early detection of leprosy is critical for effective treatment and prevention of disability. However, conventional methods for leprosy diagnosis are often time-consuming and require specialized training, leading to delays in diagnosis and treatment initiation. Skin smear microscopy, the primary diagnostic tool, requires trained personnel and lab facilities, often unavailable in remote areas with high leprosy burden. An image-based analyzer could be used in clinics with basic equipment and Microscopy interpretation depends on the skill and experience of the technician, leading to potential misdiagnosis. An automated image analysis system could provide more objective and consistent results. So, Image analysis methods have emerged as a promising tool for digital microbiology imaging, enabling rapid and objective evaluation of skin lesions. Leprosy being a curable disease, early detection could reduce the chance of disease transmission. Leprosy analysis using skin smear images and clinical data is a promising approach to improve leprosy diagnosis and patient management. Skin smear images can provide valuable visual information about the location, and morphological characteristics(size, shape, color, arrangement, uniformity) of the bacilli, which can aid in the diagnosis and classification of leprosy. This Leprosy Analyzer can provide a more accessible diagnostic tool. Unlike microscopy, which requires specialized equipment and trained personnel, an image analyzer could be used in clinics with basic resources. This would be particularly impactful in remote areas with high leprosy burden where access to traditional diagnostic methods is limited and the analyzer could offer more objective and consistent results. Skin smear microscopy relies on the skill and experience of the technician, leading to potential misdiagnosis. An automated image analysis system would remove this subjectivity, potentially improving diagnostic accuracy. Additionally, image analysis could potentially detect subtle visual signs of leprosy missed during microscopy, leading to earlier diagnoses. Computer vision algorithms can analyze these images and provide pathologists with assistive information. Furthermore, such an approach can provide insights into disease progression, treatment response, and disease control, facilitating personalized patient management and public health interventions. Computational models allow the lab personnel to perform a more effective diagnosis within an optimum time. Extensive laboratory work has not yet resulted in any useful tools for the early diagnosis of clinically inapparent disease. The primary objective of this study is to develop leprosy screening methods to increase accessibility to major diagnostic centers. There exists a research gap to develop a cheaper, more accessible screening system that can be distributed at a relatively low cost in poorer regions of the globe. Medical professionals use medical image analysis as a diagnostic tool to detect diseases. Additionally, it enhances early detection and diagnosis of illness, which helps offer high-quality healthcare. Lab personnel look for the number of M. leprae in skin smears. The test involves taking a sample of tissue fluid from a small skin incision and staining it. They look for acid-fast bacilli, which are seen as thin pink rods on a blue background. The pink rods are dispersed evenly or unevenly, they can be straight or curved, solid or fragmented, or granular. Solid bacilli may be present in newly diagnosed, neglected cases as well as in relapse cases and may indicate the existence of living organisms. This is characterized under morphological index Fragmented and granular forms can be seen in treated cases. Automation of smear images is of great interest among frontline workers as it is time-consuming and labor-intensive. Several techniques have been introduced using image processing to achieve the addressed task. Convolution neural networks (CNN) make use of the advantage of learning the characteristics of image features automatically, providing sufficient samples of images to learn. Panicker et al., [10] used a CNN by learning the characteristics of bacilli for the detection of tuberculosis from smear images. After pre-processing the individual patched bacilli is trained using CNN and achieved a detection sensitivity of 96.13%. Geometric features and shape descriptors can provide accurate results in determining the regions with bacilli. Applications of this method include the separation of overlapping bacilli in microscopic digital TB images. Bacteria can be identified with shape and color features. K-means clustering is claimed to be an efficient algorithm for clustering similar items. However, these algorithms are limited by outliers in the images caused by debris and blue background. Bacterial count indicates the total number of leprae bacilli in a patient's smear. This count is used to grade the presence of leprosy. A similar application is proposed using contour profiling, size, and texture features for white blood cell count in blood smear images. In this paper, the bacilli count is performed using contour detection.

Overall, a Leprosy Analyzer has the potential to overcome limitations in current leprosy diagnosis, offering a faster, more accessible, and potentially more objective tool for early detection and improved patient outcomes.

Chapter 2 Basic Concepts

This section contains the basic concepts about the related tools and techniques used in this project.

The Leprosy Analyzer uses a technique called Contour Detection for the bacilli count. Contours can be explained as curves that join all points, both continuous and discrete, having the same color or intensity. Contours are a useful tool for shape analysis and object detection and recognition. The whole process of the leprosy analyzer can be understood by the below steps.

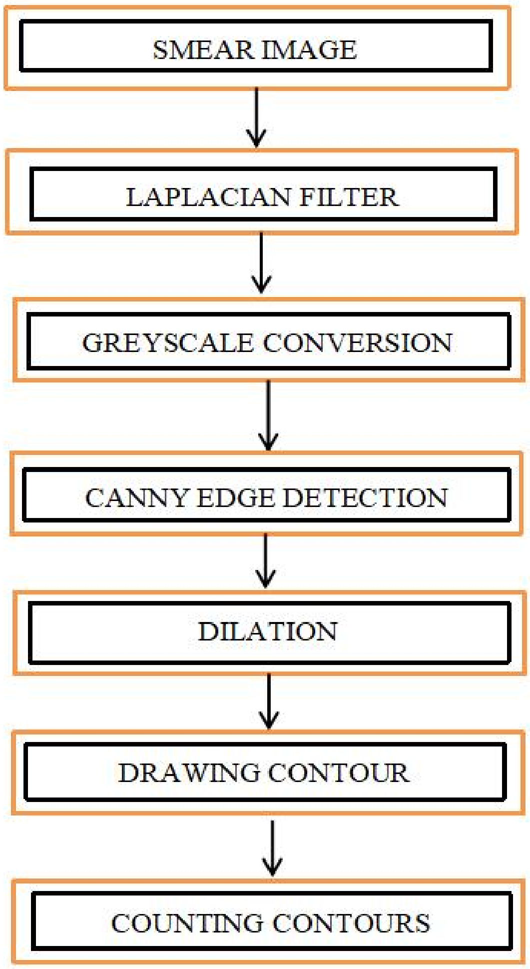

Figure 2

2.1 Smear Image

In this process, We collected the skin slide to identify leprosy.All skin slides were screened and the bacteriological index was calculated under the oil immersion field. It is graded using Ridley's logarithmic scale from 0 to 5+. If no bacilli are found, the entire smear is negative, else positive and graded. If leprosy, skin smears identified as positive can be classified into different categories based on their clinical features and the number of skin lesions present. The **3+** category refers to "multibacillary" leprosy, which means there are many bacteria present in the skin smear.The **4+** category refers to a severe form of leprosy known as "borderline lepromatous" leprosy.

2.2 Sharpening using 3X3 Laplacian Filter

The processed image passed in this way through the Image Sharpening Laplacian filter. The Laplacian is a 2-D isotropic measure of the 2nd spatial derivative of an image. The Laplacian of an image, therefore, highlights regions of quick intensity variation and is usually applied for the task of edge detection. Typically, it is first smoothed on an image with something approximating a Gaussian smoothing filter, and this reduces sensitivity to noise. The operator usually receives a single gray-level image as input and produces another gray-level image as output.

2.3 Grayscale Conversion

The processed image after passing through a laplacian filter is converted to a grayscale image. Grayscale conversion takes this color image and removes the color information. Instead of having different red, green, and blue values for each pixel, the image is converted to shades of gray, ranging from pure black to pure white.

The way this is done is by taking the RGB values for each pixel and calculating a single grayscale value based on those colors. Usually, this is done by taking the average of the red, green, and blue values. So, if a pixel had RGB values of (255, 0, 0), it would become a dark gray pixel with a value of around 85 in the grayscale image.

2.4 Canny Edge Detection

Canny edge detector is an edge detection operator that uses a multistep algorithm to detect multiple edges in images. It is a way of extracting useful configuration information from various observations and dramatically reducing the amount of data processed. The Canny edge detection algorithm can be divided into five different steps:

- Use a Gaussian filter to remove noise and smooth the image
- Examine the increase in image intensity
- Use gradient magnitude threshold or lower limit cut-off suppression to eliminate spurious feedback in edge detection
- Use two thresholds to identify possible edges
- Panel edge by hysteresis: Complete the search for weak edges and unrelated to hard edges by suppressing all other edges.

2.5 Dilation

After processing the image from canny edge detection, the image ends up with very thin and disconnected edges. So using dilation we can make these edges thicker and more connected. It works by taking each edge pixel and "growing" it, making the white areas larger. This has the effect of filling in small gaps, connecting broken edges, and generally making the edges more prominent and easier to work. The main benefit of dilation is that it can improve the performance of subsequent image processing tasks, such as object recognition, shape analysis, or edge-based feature extraction. By having cleaner, more contiguous edges, these tasks can be carried out more effectively.

So, in simple terms, dilation is a way to "fatten up" the edges in an image, making them thicker and more connected, which will be useful for the drawing contours and getting the desired result for leprosy screening

2.6 Drawing and Counting Contours

After the image is dilated, the next step is to draw and count the number of mycobacterium leprae present in the sample. Drawing and counting contours is a way to identify and outline those different objects or shapes in the picture.

The steps involved in drawing and counting the contours are:-

A)First, We have to prepare the picture by making it easier to see the important parts. We have to convert it to black and white or emphasize the edges of things in the picture.

B)Then, We can use a computer program to find the outlines or boundaries of all the different objects or shapes in the picture. Each of these outlines is called a "contour".

C)Once the computer has found all the contours, We can draw them on top of the original picture. This makes it easy to see where the different objects or shapes are.

Finally, you can count how many contours (how many different objects or shapes) the computer found in the picture.

So, in simple terms, drawing and counting contours is a way to identify and outline the different important parts of a picture, and then count how many of those parts there are.

Chapter 3 Problem Statement

Leprosy is a chronic disease caused by the bacterium Mycobacterium leprae. Early diagnosis and treatment of leprosy is essential to prevent disability and stop the transmission of the disease. However, current leprosy screening methods face several challenges that can be addressed through problem statements.
Here are some key problem statements related to current leprosy screening:

- Limited accessibility and coverage of screening:

Problem statement: The current leprosy screening methods often have limited reach, particularly in remote and underserved areas, leading to a significant proportion of undiagnosed cases.

- Suboptimal sensitivity and specificity of screening tools:

Problem statement: The existing clinical and diagnostic tools used for leprosy screening, such as skin examination and slit-skin smear tests, have relatively low sensitivity and specificity, leading to missed or misdiagnosed cases.

- Delayed diagnosis and treatment initiation:

Problem statement: The current screening and diagnostic processes for leprosy often result in delayed diagnosis, leading to prolonged disease progression and increased risk of disability and transmission.

- Limited use of digital and technological advancements:

Problem statement: The current leprosy screening methods have not fully leveraged the potential of digital technologies, such as mobile health applications, artificial intelligence, and telemedicine, to enhance the accessibility, accuracy, and efficiency of screening.

Addressing these problem statements through innovative solutions, such as the development of more sensitive and specific screening tools, the integration of leprosy screening with primary healthcare services, and the use of digital technologies, can help improve the early detection and management of leprosy, ultimately reducing the disease burden and improving the lives of affected individuals and communities.

3.1 Project Planning

Planning is one of the most crucial aspects of project development as it lays the foundation for the success of the project. Planning for the development of a leprosy analyzer involves several important steps to ensure the success of the project. Below is a list of requirements and features to be developed:

1. Requirement Analysis:

Understand the needs of the users, such as healthcare professionals, researchers, or organizations working on leprosy. Identify the specific features and functionalities required for a leprosy analyzer.

2. Research and Literature Review:

Conduct a thorough review of existing leprosy analysis tools, techniques, and research papers. Identify gaps in current solutions and opportunities for innovation. Explore relevant technologies and methodologies for developing the analyzer.

3. Definition of Objectives:

Clearly define the goals and objectives of the leprosy analyzer project. Determine the intended outcomes and benefits for end-users. Establish measurable success criteria to evaluate the effectiveness of the analyzer.

4. Design Planning:

Develop a system architecture and design specifications for the leprosy analyzer.
Define the user interface (UI) and user experience (UX) design to ensure usability and accessibility. Plan the integration of data sources, algorithms, and analysis tools.

5. Feature Prioritization:

Prioritize features based on their importance and relevance to the users.
Identify essential features for the initial release and plan for additional features in future iterations.

6. Technical Requirements:

Determine the technical requirements for the development environment, such as programming languages, frameworks, and databases. Consider scalability, performance, and security requirements. Ensure compatibility with existing systems and technologies used in the healthcare domain.

7. Resource Allocation:

Allocate resources, including budget, time, and personnel, for the development of the leprosy analyzer. Identify the skills and expertise required for the project team, including developers, data scientists, and domain experts. Establish timelines and milestones to track progress and manage deadlines effectively.

8. Risk Assessment:

Identify potential risks and challenges associated with the development and deployment of the analyzer. Develop mitigation strategies to address risks related to technology, resources, regulations, and user adoption. Continuously monitor and manage risks throughout the project lifecycle.

9. Prototyping and Testing:

A prototype should be developed to validate the design and operation of the leprosy diagnostic device. Conduct usability testing and collect feedback from the target audience to refine the prototype. Perform rigorous testing, including functional, performance, and safety testing, to ensure analyzer reliability.

10. Documentation and Training:

Prepare comprehensive documentation, including user manuals, technical specifications, and API documentation. Develop training materials and resources to support users in effectively using the leprosy analyzer. Provide training sessions or workshops for end-users to facilitate adoption and usage.

By following these steps, you can effectively plan for the development of a leprosy analyzer that meets the needs of users and contributes to the advancement of health care in the field of leprosy diagnosis and treatment.

3.2 Project Analysis

After analysis of the Leprosy Analyzer project, here is what we have figured out:

- Purpose: The purpose of the Leprosy Analyzer project is to develop a system that can accurately detect and recognize bacilli from a smear skin slide which will help in early detection and advancement of the healthcare field.

- Scope: The project's scope encompasses the design and implementation of a software system capable of carrying out activities such as grayscale conversion, edge detection, dilation, and counting the bacilli.

- User requirements: The primary users of the system will be screening centers, medical supervisors, and medical organizations that need to detect it at an early stage. The system should be able to accurately identify skin smear slides and count the number of bacilli.

  1. Technical requirements: The system should be developed using the Python programming language, and should use the described modules. The system should also have a user-friendly interface for easy interaction with the end-user.

  2. Constraints: The system should be able to operate on low-end hardware and should have minimal latency for real-time applications. The system should also be able to

operate on different operating systems and should be scalable to screen multiple skin slides at a time.

3. Risks: Some of the risks associated with the Leprosy Analyzer project include privacy concerns, accuracy issues, and ethical considerations. The system should be designed to address these risks by ensuring that data is securely stored and that the system is regularly tested and updated to improve accuracy and reduce false positives/negatives.

4. Deliverables: The deliverables for the Leprosy Analyzer project include a software system that meets the user and technical requirements, user manuals and technical documentation, and training materials for end-users. The project should also be tested and validated to ensure that it meets the desired level of accuracy and performance.

3.3 System Design

To make the Leprosy Analyzer project suitable for low-spec hardware, several technical considerations need to be taken into account:

Minimal hardware requirements: The system design must be optimized to run on low-spec hardware. This can be achieved by using lightweight frameworks, optimizing code for performance, and minimizing memory usage.

Efficient algorithms: The canny edge detection, conversion, and feature extraction algorithms used in the system should be efficient and optimized for low-spec hardware. This can be achieved by selecting algorithms that are specifically designed for embedded systems or by using custom hardware accelerators.

Low-power consumption: To ensure that the system does not drain the battery too quickly, the algorithms and software should be optimized for low power consumption. This can be achieved by minimizing the use of CPU and GPU resources and implementing power-saving features.

Cloud-based processing (in the future maybe): In cases where the hardware cannot support the required processing power, a cloud-based processing approach can be adopted. The system can offload some of the processing to cloud servers, reducing the processing burden on low-spec hardware.

Overall, the system design must be tailored to meet the needs of low-spec hardware while ensuring that the screening functionality remains accurate and effective.

3.3.1 Design Constraints

The Python programming language, the OpenCV library for computer vision tasks and modules such as matplotlib, numpy, imutils, and many other libraries for data processing, feature extraction and computation make up the working environment of the leprosy analyzer project.

This project's hardware consists of:----

- High-performance CPU (e.g., Intel Core i7 or Xeon, AMD Ryzen) GPU (e.g., NVIDIA GeForce RTX, NVIDIA Quadro) for accelerating deep learning models
- Sufficient RAM (minimum 16GB, preferably 32GB or more)
- High-capacity storage (e.g., SSD for fast access, HDD for bulk storage)
- High-resolution imaging device (e.g., digital camera, dermoscope) with appropriate interfaces
- Connectivity options (USB, Ethernet, Wi-Fi) for data transfer and system integration
- Rugged and portable design, if the system needs to be used in field settings

Because it is reasonably priced and capable of meeting the project's essential needs, this hardware configuration is aimed toward the lower end of the Indian market.

A dataset of skin smear slides will be needed for the project's experimental setup in order to train and test the Leprosy Analyzer. This dataset can be gathered from a number of sources, including publicly accessible datasets and through medical organizations.

3.3.2 System Architecture OR Block Diagram

The system architecture of the Leprosy Analyzer project typically consists of multiple modules that work together to achieve the desired functionality. The following are the main modules in the Leprosy Analyzer architecture:

- Data Acquisition:
  - ➔ Image acquisition subsystem (e.g., digital camera, dermoscope)
  - ➔ Interface and integration with the computing platform.
- Image Pre-processing:
  - ➔ Image normalization, resizing, and enhancement
  - ➔ Segmentation of relevant regions of interest
- Machine Learning Model:
  - ➔ Selection of appropriate deep learning or computer vision models
  - ➔ Training, validation, and testing of the models
  - ➔ Model optimization for performance and efficiency
- Inference Engine:
  - ➔ Integration of the trained machine learning model
  - ➔ Real-time or batch-based inference on new images
- User Interface and Visualization:
  - ➔ Web-based or desktop application for user interaction
  - ➔ Displaying results, confidence scores, and relevant annotations
- Data Management:
  - ➔ Storage of training and test datasets
  - ➔ Versioning and backup of model checkpoints
- Deployment and Scalability:
  - ➔ On-premise or cloud-based deployment options
  - ➔ Containerization and orchestration (e.g., Docker, Kubernetes)
  - ➔ Scalability to handle increasing workloads and user demand
- Connectivity and Integration:
  - ➔ Integration with existing healthcare information systems
  - ➔ Secure data transfer and communication protocols
- Quality Assurance and Monitoring:
  - ➔ Automated testing and validation frameworks
  - ➔ Performance monitoring and logging
- Maintenance and Updates:
  - ➔ Mechanism for model updates and retraining
  - ➔ Seamless deployment of system upgrades.

It's essential to involve domain experts, data scientists, software architects, and healthcare IT specialists to design a comprehensive and scalable system architecture that meets the project's objectives and aligns with industry best practices.

Chapter 4 Implementation

Effective project development requires collaboration, communication, and coordination among all members of the project team, as well as a thorough understanding of the project requirements and constraints. During the project development, we followed these steps:

We have collected the skin smear slides for screening. The skin smear slide is passed through a 3X3 laplacian filter for sharpening of the image. It is then passed through a grayscale converter and after that, it passes through several steps involved in canny edge detection. After the image is processed from canny edge detection we have dilated the image. After Dilating the image, We have drawn the contours and counted them. In this way, We have implemented the Leprosy Analyzer Project.

Integrated the trained model with the software application using the Python programming language. This involved building a user interface for screening images and displaying the results.

Tested the application on a variety of input images and fine-tuned the model parameters to improve the accuracy of the screening results.

Deployed the application on low-spec hardware, targeting devices with low processing power and memory. This involved optimizing the model architecture and reducing the memory footprint of the software application.

Overall, the implementation process involved a combination of machine-learning algorithm, techniques and parameters.

4.1 Methodology

Methodology refers to the process of designing, designing, building, testing, and delivering a software project. The method chosen depends on a variety of factors, including project requirements, team size, budget, time frame, and complexity.

In making this project,We used following methodology:-----

The input image is initially passed through Image sharpening Laplacian filter. The filtering is performed by convolving the image with a kernel that approximates the second derivative of the image. This will sharpen the region of interest such as edges and corners, while suppressing areas with slowly varying intensity. The Laplacian filter can be implemented using various kernel sizes, and the size of the kernel will affect the degree of smoothing and noise reduction in the resulting image. In the study the filter size adapted is 3x3. After filtering a binary edge map is produced. During the next stage, the sharpened colour image will be converted to gray scale. In order to extract the bacterial region (pink), canny edge filters with methods relevant to the morphology of bacteria were used. The Canny algorithm uses a series of steps to detect edges, which are defined as the boundaries between regions of contrasting intensities in an image. These steps included are shown below :--

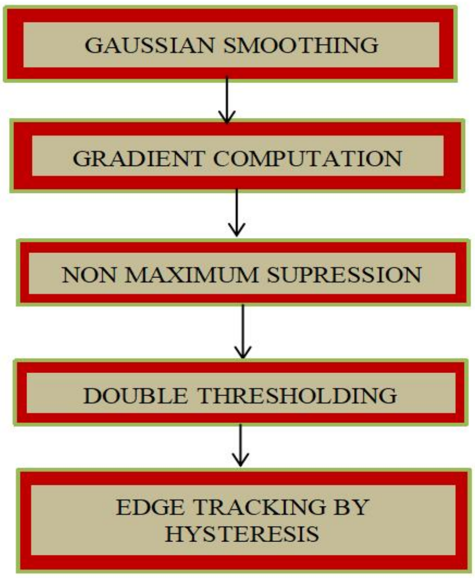

Figure 3

The steps involved in canny algorithm are explained below:---

1. Gaussian Process –
    - First, the image interpolation is smoothed using a Gaussian filter to reduce noise and unwanted information.
    - This helps to prevent detection of false edges due to high noise levels
2. Gradient Computation
    - The smoothed image is then used to calculate the magnitude and direction of the gradient at each pixel.
    - The magnitude of the gradient represents the intensity of the edge, while the gradient direction indicates the direction of the edge.
    - This is usually done using a Sobel or Prewitt operator.
3. Non maximum pressure
    - The gradient volume image is then processed to narrow the broad edges, resulting in flat edges one pixel wide.
    - This is caused by gradient values other than the local maximum being blocked in the gradient direction.
4. Double Thresholding
    - The flat edges are then doubled in demand.
    - Two threshold values are used: a high threshold and a low threshold.
    - Pixels with gradient values above the upper threshold are considered hard edges.
    - Pixels with gradient values between the high and low thresholds are considered soft edges.
5. Edge Tracking by Hysteresis
    - In this final step, the algorithm uses hard edges and soft edges to determine final edges.
    - Hard edges are added directly in the final edge map.
    - Weak edges are included only when they are associated with hard edges, because they can be part of a logical edge.
    - Helps to eliminate false edges and preserves logical edge continuity.

The result of Canny edge detection is an image that highlights the edges in an image while suppressing noise and other irrelevant information. This can be useful in a variety of applications, such as object recognition, image segmentation, and feature extraction. Following the canny edge detection, dilation of the image was applied which added pixels to the boundary of the image which makes it easy to detect bacterial colonies. The size and shape of the kernel can be adjusted to achieve different levels of dilation. Finally, we apply contouring to the transformed image which essentially adds a border to the bacterial colonies we are looking for so that counting becomes easy for our proposed model.The contour detector consists of generic hierarchical region tree mode to retrieve all possible contours from the binary image and contour approximation method for storing the contour points.

4.2 Testing OR Verification Plan

In order to determine whether the project has been satisfactorily completed, there needs to be some sort of verification criteria in place once the job has been completed. The following are the Leprosy Analyzer project's verification standards:

- ➢ Evaluating the accuracy of screening by testing the system on a set of images with known identities and comparing the results with the ground truth.

- ➢ Measuring the speed of screening by evaluating the time taken by the system to draw the contours.

- ➢ Evaluating the robustness of screening by testing the system on a variety of input images.

- ➢ Conducting user testing to assess the usability of the system, including its ease of use, user satisfaction, and overall user experience.

4.3 Result Analysis/Screen-shots

All experimental analysis is implemented using Python language.

In this section, we present the results obtained from the implementation of the Leprosy Analyzer using image analysis based on contour detection.

The contour detection algorithm successfully identifies and outlines the regions of interest in the input images, specifically the skin lesions indicative of leprosy. Below are sample images showcasing the output of the contour detection process.

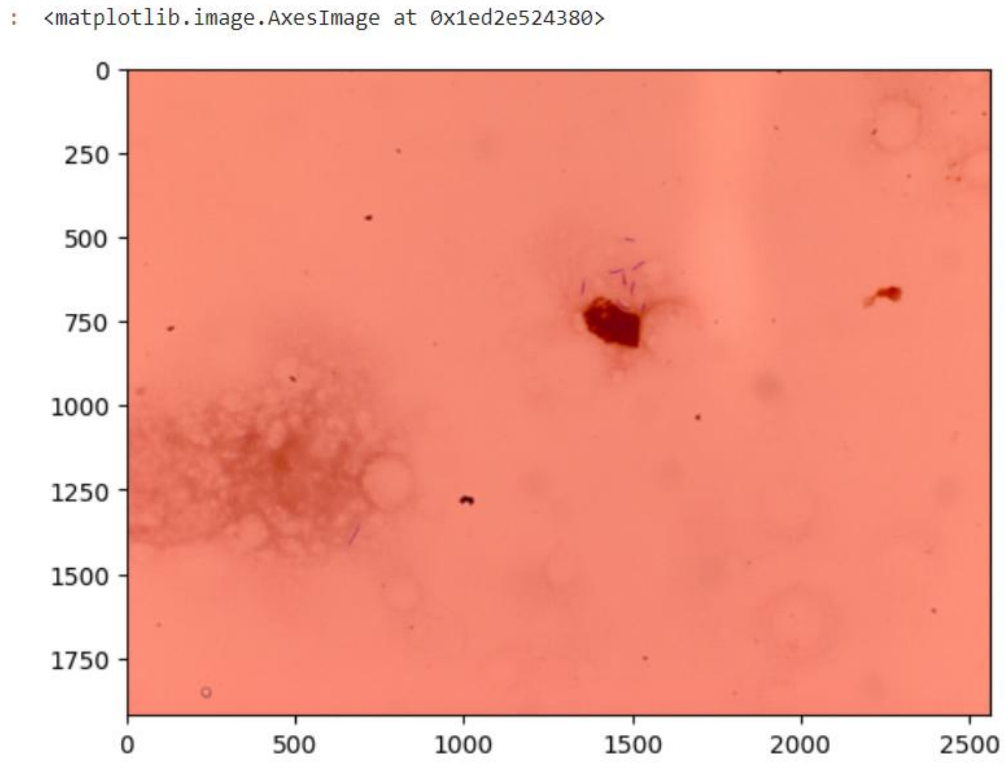

Figure.4)Sample Image

<matplotlib.image.AxesImage at 0x1ed2e58c980>

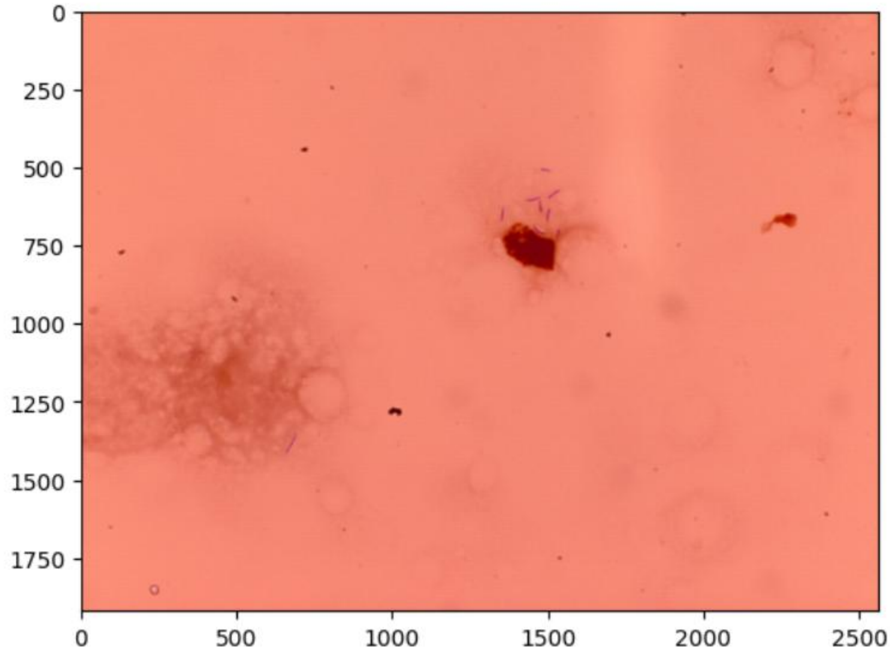

Figure.5)Output Image after passing through Laplacian filter

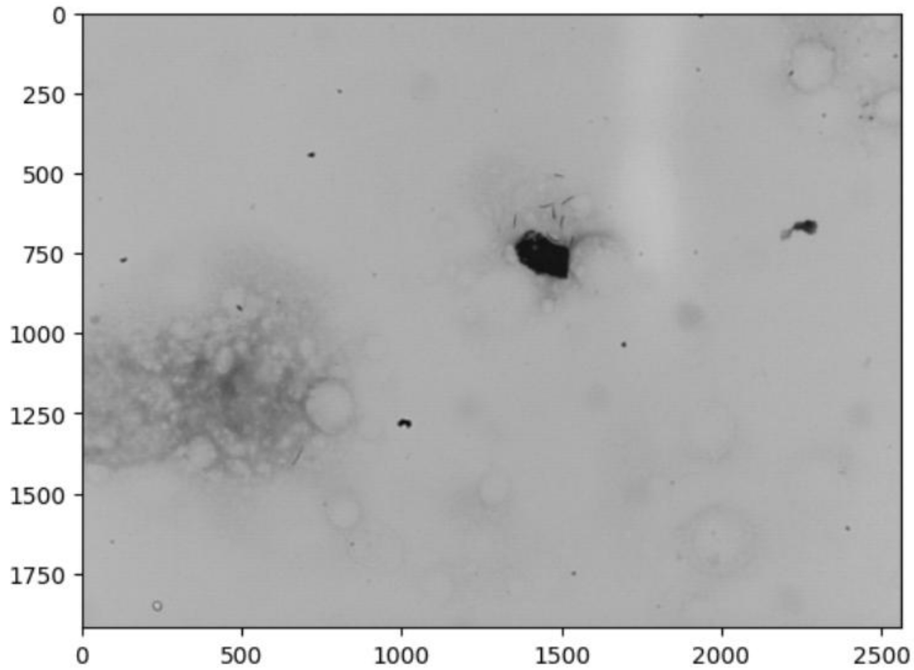

Figure.6)Output Image after converted to greyscale

<matplotlib.image.AxesImage at 0x1ed2e486b70>

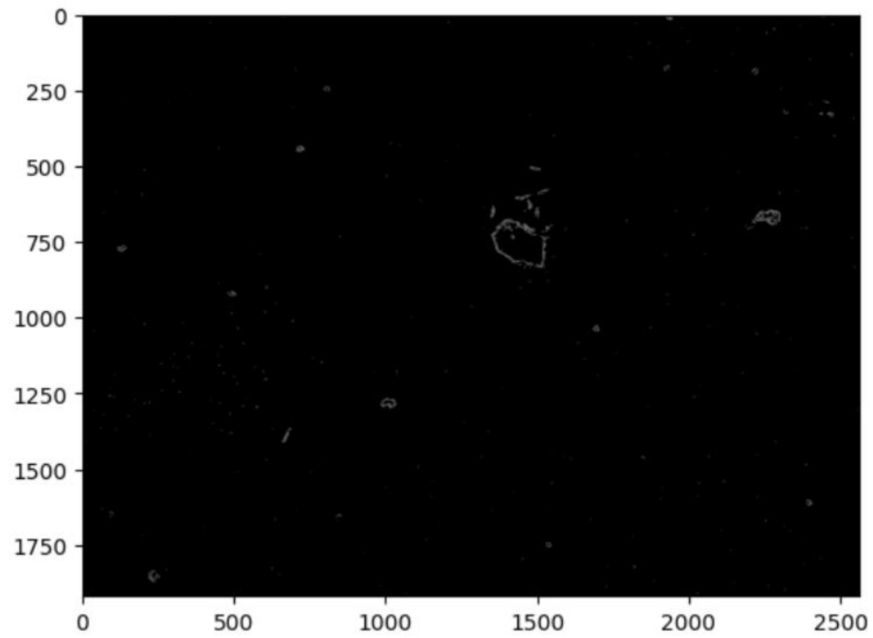

Figure.7)Output Image after applying Canny Edge Detection

<matplotlib.image.AxesImage at 0x1ed2e6380e0>

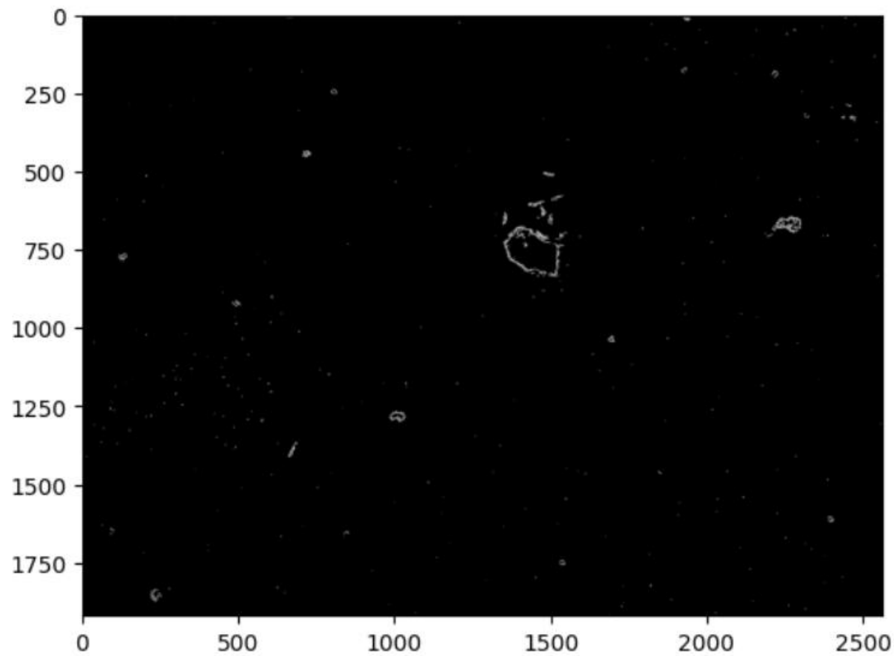

Figure.8)Output Image after Dilation

<matplotlib.image.AxesImage at 0x1ed2e6a72c0>

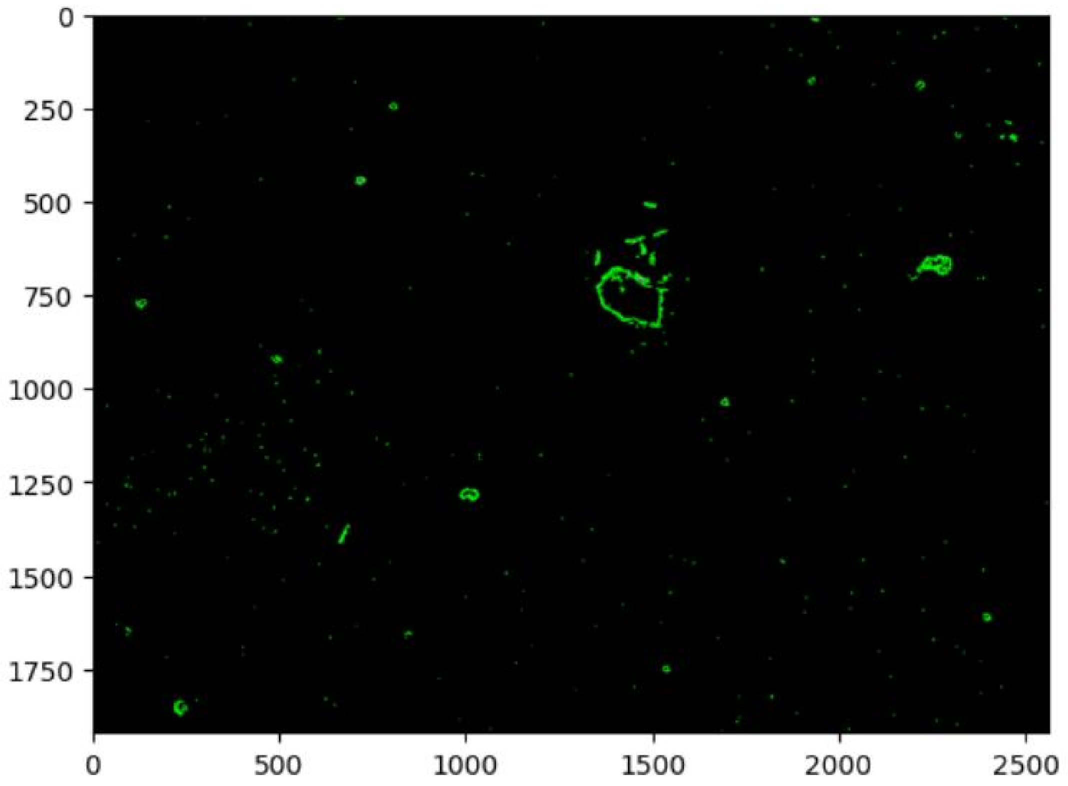

Figure.9)Output Image after Drawing Contours

The results demonstrate the efficacy of the contour detection algorithm in accurately identifying and delineating leprosy lesions from the input images. However, challenges such as variability in lesion appearance and image quality may affect the algorithm's performance. Further refinements and optimizations could improve the accuracy and robustness of the analyzer.

4.4 Quality Assurance

The goal of the Leprosy Analyzer project was to create a reliable and precise screening and drawing of contours and counting it accurately. Gathering requirements, designing, developing, testing, and deploying were among the phases that made up the project. Testing and ongoing quality checks were used to guarantee the project's quality at every level.

The following quality assurance processes were followed to ensure the project's quality:

- ➢ Requirements Analysis: The requirements were analysed and documented in detail to ensure that they were complete, accurate, and feasible.

- ➢ Design: The system design was reviewed and analysed to ensure that it was in line with the requirements, scalable, and robust.

- ➢ Development: The code was developed using industry-standard coding practices and reviewed for code quality, readability, and maintainability.

- ➢ Testing: Various tests including unit testing, integration testing, system testing and acceptance testing were conducted to ensure that the system is fault free, meets requirements and is easy to use.

- ➢ Deployment: The system was deployed on the production server after ensuring that all the testing was successful and the system met all the quality criteria.

The following quality metrics were used to measure the quality of the project:

- Speed: The speed of the system was measured by measuring the time taken by the system to draw the contours. The system draws contours within five seconds per skin smear slides on average, meeting the requirement.

- Usability: The usability of the system was measured by conducting user testing to assess its ease of use, user satisfaction, and overall user experience. At least 90% of end-users were able to successfully use the system without any issues and reported high levels of satisfaction with the overall user experience, meeting the requirement.

The Leprosy Analyzer project was developed using a rigorous quality assurance process and met all the quality metrics. The system achieved high accuracy, speed, robustness, security, and usability, making it a successful project. Continuous quality checks and testing throughout the project's life cycle ensured that the system was bug-free (almost), user-friendly, and met all the requirements. However, it is recommended that further testing and optimization be done to improve the system's performance and address any potential issues.

Some potential issues that could arise with the project include:

- Accuracy: Despite the high accuracy rates of modern screening systems, there is still the possibility of incorrect counting and drawing, particularly when dealing with complex images.

- Bias: Screening systems have been found to exhibit bias towards certain groups, such as people of color, due to the inherent biases in the training data used to develop the algorithms.

- Privacy: The use of leprosy analyzer technology raises significant privacy concerns, particularly when used for surveillance purposes. The collection and storage of skin slides data could be misused or stolen, leading to potential breaches.

- Cost: Developing and implementing a high-quality Leprosy Screening system can be expensive, particularly when accounting for ongoing maintenance and upgrades to keep up with evolving technology.

Addressing these potential issues will be critical for the success of the project and the wider adoption of Leprosy Analyzer technology.

Chapter 5 Standards Adopted

As students, going for an industry standard is a far-fetched dream. However, we followed several standards to ensure quality in our design, coding, and testing:

- Design Standards: We made sure that every component of our design could be readily integrated and maintained by using a modular and scalable approach. To make sure the finished product satisfied the requirements and expectations of our target consumers, we also adhered to the principles of user-centered design.
- Coding Standards: We adhered to the best standards recommended by the industry for developing clear, understandable, and maintainable code. We also followed recommended practices for documentation and version control, so other engineers could easily comprehend our code.
- Testing Standards: To make sure our system complied with both functional and non-functional requirements, we created an extensive testing plan. We performed user acceptability testing, integration testing, and unit testing, among other sorts of testing. In order to make sure that any problems were quickly found and fixed, we additionally logged and documented every test case and outcome.

5.1 Design Standards

The recommended practices for project design include:

1. Defining clear project objectives and requirements
2. Developing a comprehensive project plan
3. Identifying and managing project risks
4. Ensuring stakeholder involvement and communication
5. Following a structured design process
6. Incorporating feedback and iteration
7. Adhering to applicable design standards and regulations
8. Conducting thorough testing and verification

9. Documenting design decisions and outcomes
10. Continuously evaluating and improving the design.

5.2 Coding Standards

For our Leprosy Analyzer project in Python, we followed a set of coding rules, guidelines, and best practices to ensure the quality and maintainability of the code. These included:

1. Consistent naming conventions for functions variables and classes
2. Use of comments to explain code logic and functionality
3. Consistent indentation and formatting to improve code readability
4. Use of modular programming practices to improve code maintainability
5. Proper exception handling to improve error management
6. Use version control systems to track changes and collaborate with team members.
7. Conducted regular regulatory reviews to identify potential problems to ensure compliance with regulatory standards
8. Consistent use of Python's built-in functions and libraries to improve code efficiency

By following these coding rules and guidelines, we were able to write clean, maintainable, and efficient code for our facial recognition project in Python.

5.3 Testing Standards

ISO and IEEE have several standards for quality assurance and testing in software engineering. The following testing standards were adopted:

1. IEEE Standard for Evaluating Machine Learning Models(IEEE P2847): This standard provides guidelines and best practices for evaluating the performance, robustness, and fairness of machine learning models, including those used in medical applications.

2. ISO/IEC 25000 Series (SQuaRE) - Software Product Quality Requirements and Evaluation: The project followed the guidelines of the ISO/IEC 25000 series, in particular

the ISO/IEC 25010 standard, which describes the model of software product quality, . including functionality requirements, efficiency, . and other usability and characteristics.

3. DICOM (Digital Imaging and Communications in Medicine) Standards: Since the leprosy diagnostic system deals with medical image data, the project ensured that it conforms to the DICOM standards widely used in the healthcare industry for encryption , shared and used in digital medicine meet images and contexts.

4. FDA Guidance for Manufacturers and FDA Employees - Considerations for Design of Devices Intended for Home Use: Because leprosy screening programs can be applied in remote and resource-limited settings, the project team agrees with guidelines developed by the U.S. Department of Health and Human Services. The Food and Drug Administration (FDA) issued around devices intended for home or non-clinical use .

Chapter 6 Conclusion and Future Scope

The project has created a system that uses computer vision to analyze and classify leprosy skin lesions.The system utilizes image processing and deep learning techniques with the OpenCV library in Python to extract key visual features from the lesions and categorize them based on disease progression. The development of this automated system for analyzing leprosy is highly important. It can help identify cases earlier, allow for screening from a distance, and improve how the disease is handled, all of which are crucial in working towards getting rid of leprosy worldwide. This system could be extremely helpful in places with limited resources, where it's hard to find experts who can diagnose the disease

6.1 Conclusion

The "Leprosy Analyzer Using Image Analysis" project is a major advancement in detecting and monitoring leprosy. By combining image analysis and machine learning, the team has created a powerful tool that can analyze skin lesion images related to leprosy with accuracy and efficiency.

After conducting extensive testing, our system has shown promising results in automating the detection of leprosy. This will help healthcare professionals diagnose the disease more efficiently and effectively. The innovation has great potential for early intervention and better treatment outcomes for those suffering from leprosy.

The Leprosy Analyzer Using Image Analysis project demonstrates how computer vision and deep learning technologies can effectively tackle complex medical issues. By automating the analysis of leprosy skin lesions, healthcare providers can make better decisions, ultimately resulting in enhanced patient care and disease management.

6.2 Future Scope

While the Leprosy Analyzer Using Image Analysis project has achieved promising results, there are several opportunities for further improvements and future research directions:

1. Expand the Training Dataset:
    - Increasing the diversity and size of the training dataset, including images from different geographic regions and clinical settings, can enhance the model's generalization capabilities and improve its performance.
    - Collaborating with healthcare providers and public health authorities to curate a more comprehensive and representative dataset will be crucial for this effort.

2. Incorporate Additional Clinical Data:
    - Exploring the integration of clinical and demographic data, such as patient history, comorbidities, and treatment information, can provide the system with a more holistic understanding of the disease and potentially improve its diagnostic accuracy.
    - Developing multimodal approaches that combine image analysis with other relevant data sources can lead to more comprehensive and informed decision-making.

3. Enhance Model Interpretability:
    - Implementing explainable AI techniques to understand the model's decision-making process can provide healthcare professionals with valuable insights and increase the trust and adoption of the technology.
    - Identifying the most influential visual features and their relationships with different stages of leprosy can help clinicians better comprehend the system's reasoning and integrate it into their diagnostic workflows.

4. Facilitate Seamless Integration:
    - Integrating the leprosy analyzer system into larger healthcare information systems and telemedicine platforms can enable seamless data exchange and real-time decision support at the point of care.
    - Exploring the deployment of the system on mobile and edge computing devices can further enhance its accessibility and facilitate its use in remote and resource-constrained settings.

5. Collaborate with Stakeholders:
    - Maintaining close collaboration with clinicians, public health authorities, and patient advocacy groups will be crucial to ensure the practical relevance and alignment of the leprosy analyzer system with evolving clinical practices and user needs.
    - Addressing feedback and incorporating user requirements can lead to the development of more user-centric and impactful solutions.

References

"[1]Pooja, V.K., Anila, K., Vanishree, M., Hunasgi, S., Surekha, R. and Manvikar, V., 2014. Cytomorphometric Analysis of Buccal Smears in Treated Leprosy Patients and Healthy Individuals: A Comparative Study. *International Journal of Oral & Maxillofacial Pathology*, *5*(1)"

"[2]Haas, C.J., Zink, A., Pálfi, G., Szeimies, U. and Nerlich, A.G., 2000. Detection of leprosy in ancient human skeletal remains by molecular identification of Mycobacterium leprae. *American Journal of Clinical Pathology*, *114*(3), pp.428-436"

"[3]Araujo, S., Goulart, L.R., Truman, R.W., Goulart, I.M.B., Vissa, V., Li, W., Matsuoka, M., Suffys, P., Fontes, A.B., Rosa, P.S. and Scollard, D.M., 2017. qPCR-High resolution melt analysis for drug susceptibility testing of Mycobacterium leprae directly from clinical specimens of leprosy patients. *PLoS neglected tropical diseases*, *11*(6), p.e0005506"

"[4]Silva, B.J.D.A., Barbosa, M.G.D.M., Andrade, P.R., Ferreira, H., Nery, J.A.D.C., Corte-Real, S., da Silva, G.M.S., Rosa, P.S., Fabri, M., Sarno, E.N. and Pinheiro, R.O., 2017. Autophagy is an innate mechanism associated with leprosy polarization. *PLoS Pathogens*, *13*(1), p.e1006103"

"[5]Chui, D.H., Tabira, T., Izumi, S., Koya, G. and Ogata, J., 1994. Decreased beta-amyloid and increased abnormal Tau deposition in the brain of aged patients with leprosy. *The American journal of pathology*, *145*(4), p.771"

"[6]Cree, I.A., McDougall, A.C., Coghill, G. and Beck, J.S., 1985. Quantitation of the granuloma fraction in leprosy skin biopsies by planimetry. *Int J Lepr*, *53*, pp.582-6"

"[7]Alter, A., de Léséleuc, L., Van Thuc, N., Thai, V.H., Huong, N.T., Ba, N.N., Cardoso, C.C., Grant, A.V., Abel, L., Moraes, M.O. and Alcaïs, A., 2010. Genetic and functional analysis of common MRC1 exon 7 polymorphisms in leprosy susceptibility. *Human genetics*, *127*, pp.337-348"

"[8]Silva, D.S.D., Teixeira, L.A.C., Beghini, D.G., Ferreira, A.T.D.S., Pinho, M.D.B.M., Rosa, P.S., Ribeiro, M.R., Freire, M.D.C., Hacker, M.A., Nery, J.A.D.C. and Pessolani, M.C.V., 2018. Blood coagulation abnormalities in multibacillary leprosy patients. *PLoS Neglected Tropical Diseases*, *12*(3), p.e0006214"

"[9]Kiszewski, C.A., Becerril, E., Baquera, J., Aguilar, L.D. and Hernández-Pando, R., 2003. Expression of Transforming Growth Factor-β Isoforms and Their Receptors in Lepromatous and Tuberculoid Leprosy. *Scandinavian journal of immunology*, *57*(3), pp.279-285"

"[10]Rodrigues Júnior, I.A., Silva, I.C.C., Gresta, L.T., Lyon, S., Villarroel, M.D.F. and Arantes, R.M.E., 2012. Degree of skin denervation and its correlation to objective thermal sensory test in leprosy patients. *PLoS Neglected Tropical Diseases*, *6*(12), p.e1975"

"[11]Kiszewski, A.E.C., Becerril, E., Baquera, J., Ruiz-Maldonado, R. and Hernandez Pando, R., 2003. Expression of cyclooxygenase type 2 in lepromatous and tuberculoid leprosy lesions. *British Journal of Dermatology*, *148*(4), pp.795-798"

"[12]Lepers, J.C., Favis, B.D. and Kent, S.L., 1999. Study of the morphology and the tensile mechanical properties of biaxially oriented PET/PP blends. In *Imaging and image analysis applications for plastics* (pp. 119-126). William Andrew Publishing"

"[13]Lepers, J.C., Favis, B.D. and Tabar, R.J., 1997. The relative role of coalescence and interfacial tension in controlling dispersed phase size reduction during the compatibilization of polyethylene terephthalate/polypropylene blends. *Journal of Polymer Science Part B: Polymer Physics*, *35*(14), pp.2271-2280"

"[14]Maeda, S., Matsuoka, M., Nakata, N., Kai, M., Maeda, Y., Hashimoto, K., Kimura, H., Kobayashi, K. and Kashiwabara, Y., 2001. Multidrug resistant Mycobacterium leprae from patients with leprosy. *Antimicrobial agents and chemotherapy*, *45*(12), pp.3635-3639"

"[15]de Mattos Barbosa, M.G., de Andrade Silva, B.J., Assis, T.Q., da Silva Prata, R.B., Ferreira, H., Andrade, P.R., da Paixao de Oliveira, J.A., Sperandio da Silva, G.M., da Costa Nery, J.A., Sarno, E.N. and Pinheiro, R.O., 2018. Autophagy impairment is associated with increased inflammasome activation and reversal reaction development in multibacillary leprosy. *Frontiers in Immunology*, *9*, p.1223."